

UNIVERSITÉ DE MONTRÉAL
FACULTÉ DES ARTS ET DES SCIENCES
DIRECTION DE L'ENSEIGNEMENT DE SERVICE EN INFORMATIQUE

Examen Intra
Date: jeudi 31 octobre 2002
Durée: 2:00 heures (de 16h30 à 18h30)

IFT-1969
Automne 2002
Par Michel Reid

- Documentation (non électronique) permise
- Barème: 20% de la note finale du cours.
- Vous devez répondre à toutes les questions sur le questionnaire d'examen.
- L'utilisation de téléphone cellulaire ou de téléavertisseur (paget) sont interdits, veuillez fermer ces appareils si vous en avez.

Nom :

signature :

code permanent :

Section I (Vrai ou Faux) (15 points)

Répondez par **Vrai** (ou **V**) si l'énoncé est *correct* ou par **Faux** (ou **F**) si l'énoncé est *incorrect*.

Si vous répondez Faux (et seulement dans ce cas), expliquez brièvement votre réponse. Chaque question **vaut trois (3) points**. L'explication **appropriée** (lorsque exigée) vaut un (1) point sur trois.

A. Soit les déclarations suivantes :

```
int *ptr, tab[50];
```

L'instruction suivante est valide

```
ptr = &tab[3];
```

Réponse :

vrai

B. Soit les déclarations suivante : **int i = 2, a = 10, b;**

Après l'instruction suivante :

```
b = i-- + a;
```

la valeur de b est **12**

Réponse :

vrai

C. Soit les déclarations des fonctions et des variables suivantes :

```
float moy(float tab[], int n);
void affPlusGrandMoyenne(float tab[], int n, float moy);
float leTableau[20];
int nbElements;
```

L'appel suivant est valide

```
affPlusGrandMoyenne(leTableau, nbElements, moy(leTableau,nbElements)) ;
```

Réponse :

vrai

D. La déclaration de la fonction suivante est valide :

```
float fonction(float tab[][] , int n, int m);
```

Réponse :

Faux, on doit spécifier la deuxième borne du tableau

E. La déclaration suivante est valide :

```
int n = 25; , unTableau[n];
```

Réponse :

Faux : après le ';' il ne peut y avoir de ',' et la taille du tableau ne peut être n qui est une variable

Section 2 : Erreurs de compilation et de logique (15 points)

Le programme suivant permet de saisir la position d'un pilote à la fin d'une course de Formule 1 et d'afficher un ou plusieurs messages pertinents en utilisant un `switch`. Le programme fonctionne pour plusieurs pilotes tant que l'utilisateur décide de continuer. Par contre, quelques erreurs se sont glissées dans le programme.

Vous devez donner quatre (4) erreurs de syntaxe (il y en a environ 10) et, pour chacune, indiquer le numéro de la ligne où elle apparaît et la corriger.

De plus, trouvez une erreur de logique qui résulte en un comportement erroné à l'exécution, indiquez les corrections que vous feriez et sur quelle(s) ligne(s).

Note: les numéros de lignes ne font pas partie du programme, ils ne servent que pour faire référence aux lignes du programme.

```
1. #include <stdio.h>
2. #include <ctype.h>

3. void main()
4. { int position ,
5.   char Reponse = 'n' ;

6.   do
7.   { printf("Entrez votre position à l'arrivée : ");
8.     fflush(stdin); /* effacer la mémoire temporaire */
9.     scanf("%d", &position);

10.    switch(position) ;
11.    {
12.      case 1 : printf("Bravo vous êtes le vainqueur! "\n);
13.      case 2 :
14.      case 3 : printf("vous avez mérité la place numéro %d sur"
15.                  " le podium et\n", ) ;
16.      case 4 || 5 :
17.      case 6 : printf("Vous avez terminé dans les points");
18.              break;
19.      case 7 : printf("Vous n'avez manqué les points que d'une
20.                  position!");
21.              break ( ) ;
22.      else : printf("Vous ne récoltez aucun point");
23.    }

24.    printf("\nVoulez-vous continuer ? (O/N) " );
25.    fflush(stdin);
26.    scanf("%c", Reponse);
27.  } while (Reponse = 'o')
28. }
```

Réponse: (pour chaque erreurs trouvées, indiquez le numéro de la ligne et la correction nécessaire)

compilation

logique

Section 3 : Simulation (15 points)

```
void mystere(int n)
{
    int i,j;

    printf("pour n = %d\n\n", n);

    if(n%2 ==1)
    {
        for(i=0; i<n; i++)
        {
            for( j=0; j<n; j++)
                if( i==j || i == n-1-j )
                    printf("*");
            else
                printf(" ");
            printf("\n");
        }
    }
    printf("\n\nfin du mystere\n");
}
```

a) (10 points) Que fait afficher l'appel suivant : **mystere(3);**

pour n = 3

```
* *
 *
* *
```

fin du mystere

b) (5 points) Que fait afficher l'appel suivant : **mystere(4);**

pour n = 4

fin du mystere

Section 4 : Expliquez le traitement de la fonction suivante (15 pts)

```
int mystere(char M[], int n)
{
    int i, c=0 ;
    for(i=0; i<n; i++)
        if(M[i]<= 'Z' && M[i] >= 'A')
            c++;
    return c;
}
```

Expliquez, en français, le traitement général de cette fonction :

Compte et retourne le nombre de lettres majuscules dans un tableau de caractères.

Section 5 : Programmation (40 points)

a) Écrivez la fonction suivante qui effectue le produit d'un vecteur (tableau) et une matrice (tableau à 2 dimensions).

```
int prodVect(int M[][MAX],int vect[],int nb, int ln, int col, int R[]);
```

M est un tableau à 2 dimensions de `ln` lignes et `col` colonnes.

`vect` est un tableau de `nb` éléments.

R est le tableau résultant.

le nombre d'éléments de R sera retourné par la fonction.

Traitement : Un vecteur `vect` de `n` éléments qui multiplie une matrice M de `n` x `m` donne un vecteur R de `m` éléments.

$$R_j = \sum_{i=0}^{n-1} (vect_i * M_{ij}) \quad ? j \text{ tel que } 0 \leq j \leq m-1$$

Réponse :

```
int prodVect(int M[][MAX],int vect[],int nb, int ln, int col, int R[]){
```

```
    int i,j,somme;
```

```
    if(nb != ln) return 0;
```

```
    for(j=0;j<col; j++)
```

```
    {
```

```
        somme = 0;
```

```
        for(i=0;i<ln;i++)
```

```
            somme += (vect[i]*M[i][j]);
```

```
        R[j]= somme;
```

```
    }
```

```
    return col;
```

```
}
```

- c) Écrivez une fonction qui transmet comme résultat un tableau (de réels) contenant les moyennes de chacune des colonnes d'une matrice d'entiers. La fonction reçoit en paramètres la matrice, son nombre de lignes, de colonnes ainsi que le tableau à remplir. Celle-ci retourne le nombre d'éléments du tableau.

Réponse :

```
int tabMoy(int M[][MAX], int ln, int col, float[])
{

    int i,j;
    float somme;

    if(ln == 0) return 0;
    for(j=0;j<col; j++)
    {
        somme = 0;
        for(i=0;i<ln;i++)
            somme+= M[i][j];
        R[j]=somme/ln;
    }
    return col;
}
```

}